

BUNDESREPUBLIK DEUTSCHLAND



REC'D 27 MAR 2000

WIPO PCT

4
D 00 / 270**Bescheinigung**

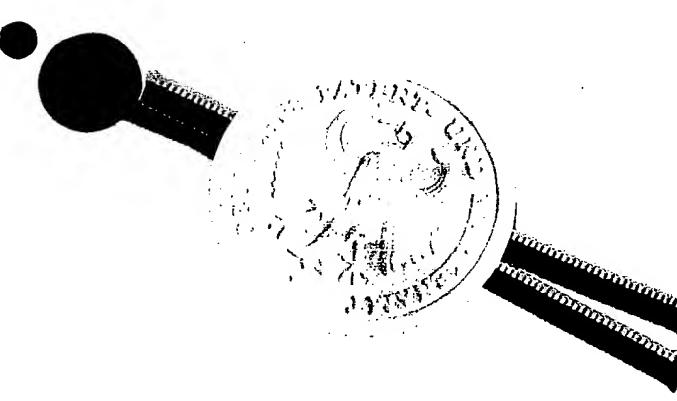
Die Siemens Aktiengesellschaft in München/Deutschland hat eine Patentanmeldung unter der Bezeichnung

"Verfahren und Vorrichtung zum Berechnen einer diskreten
orthogonalen Transformation wie FFT oder IFFT"

am 18. Februar 1999 beim Deutschen Patent- und Markenamt eingereicht.

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

Die Anmeldung hat im Deutschen Patent- und Markenamt vorläufig das Symbol G 06 F 17/14 der Internationalen Patentklassifikation erhalten.



München, den 14. März 2000

Deutsches Patent- und Markenamt

Der Präsident

Im Auftrag



Nietiedt

Aktenzeichen: 199 06 868.2

**PRIORITY
DOCUMENT**

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)



Beschreibung

Verfahren und Vorrichtung zum Berechnen einer diskreten orthogonalen Transformation wie FFT oder IFFT

5

Die Erfindung betrifft ein Verfahren und eine Schaltung zur Berechnung der Fast-Fourier-Transformation sowie der dazu inversen Transformation, im folgenden als FFT und IFFT bezeichnet, oder ähnlich strukturierter Transformationen, d.h. diskrete orthogonale Transformationen.

10

Die FFT und IFFT sind in der Nachrichtentechnik äußerst wichtige Transformationen, da sie beispielsweise die Beschreibung eines Sachverhalts im Zeitbereich in eine solche im Frequenzbereich transformieren und umgekehrt.

15

In der digitalen Signalverarbeitung wird häufig eine sog. N-Punkt diskrete Fourier-Transformation, im folgenden als DFT bezeichnet, berechnet, die wie folgt definiert ist:

20

$$X(k) = \sum_{n=0}^{N-1} x(n) w^{nk} \quad k = 0, 1, \dots, N-1$$

mit $w = e^{-j} 2\pi/N$

Die Komplexität der Berechnung der DFT ist proportional zu $O(N^2)$. Durch die Anwendung der FFT kann die Komplexität der Berechnung auf $O(N \log(N))$ reduziert werden. Dies geschieht durch eine hierarchische Aufteilung der Berechnung in Transformationen kürzerer Folgen.

30

Zur Berechnung der FFT existieren zwei grundlegende Algorithmen. Der eine wird "Decimation in Frequency" (DIF), der andere "Decimation in Time" (DIT) genannt. Im folgenden wird beispielhaft der DIT-Algorithmus behandelt.

35

hier der Zugriff auf die einzelnen Zellen uneingeschränkt möglich ist.

5 Daher wird die Geschwindigkeit der FFT vor allem durch die Schnittstelle zum Speicher begrenzt.

Der Erfundung liegt daher die Aufgabe zugrunde, ein Verfahren und eine Vorrichtung zur Berechnung diskreter orthogonaler Transformationen, insbesondere der FFT und IFFT, zu schaffen, 10 womit eine schnellere Berechnung möglich ist.

Die Aufgabe wird durch die Merkmale der unabhängigen Ansprüche gelöst. Bevorzugte Ausgestaltungen der Erfundung sind Gegenstand der Unteransprüche.

15 Das erste erfundungsgemäße Verfahren zum Berechnen einer orthogonalen diskreten Transformation nach dem DIT-Verfahren in vorgegebenen Zwischenschritten, wobei unter einem Zwischenschritt die Addition sowie Multiplikation einer Verarbeitungsstufe zu verstehen ist, umfaßt die folgenden Schritte:

- a) Lesen der Daten aus einem seitenweise organisierten Speicher,
- b) Durchführen des durch die Transformation vorgegebenen Zwischenschritts,
- c) Speichern der resultierenden Daten in einem Zwischenspeicher, und
- d) Seitenweises Schreiben der resultierenden Daten aus dem Zwischenspeicher in den seitenweise organisierten Speicher.

30 Das erfundungsgemäße Verfahren zum Berechnen einer orthogonalen diskreten Transformation nach dem DIF-Verfahren in vorgegebenen Zwischenschritten weist die folgenden Schritte auf:

- a) Lesen der Daten aus einem seitenweise organisierten Speicher,
- b) Speichern der Daten in einem Zwischenspeicher,
- c) Durchführen des durch die Transformation vorgegebenen Zwischenschritts, und

Fig. 1 zeigt das Schema der Berechnung einer "in place"-FFT,

Fig. 2 zeigt den Signalflußgraphen einer "in place"-FFT für

5 N = 8,

Fig. 3 zeigt den Signalflußgraphen einer "Singleton"-FFT für

N = 8,

10 Fig. 4 zeigt eine erste Ausführungsform zur Berechnung einer FFT, und

Fig. 5 zeigt eine zweite Ausführungsform zur Berechnung einer FFT nach dem DIT-Verfahren.

15

Fig. 1 zeigt die oben bereits erwähnte übliche "in-place"-Berechnung einer FFT, wobei aus einem Speicher 1, üblicherweise eine DRAM ein Datenpaar gelesen wird, in einer Butterfly-Einheit 2 verknüpft und wieder in den Speicher 1 geschrieben werden.

20

Fig. 2 zeigt den grundlegenden DIT-Algorithmus einer FFT-Berechnung für N = 8. Die Daten $x(0)$ bis $x(7)$ müssen zur Berechnung am Anfang "bit-reversed" im Speicher vorliegen. Dargestellt ist der Signalflußgraph zur Berechnung der 8 Fourierkomponenten $X(0) - X(7)$. Der Algorithmus sowie der Signalflußgraph ist A.V.Oppenheim, R.W.Schafer: "Digital Signal Processing", Prentice-Hall Inc., Englewood Cliffs, New Jersey, USA, 1975, S. 285 ff, beschrieben, so daß hier nicht näher darauf eingegangen werden braucht. Hinsichtlich der Nomenklatur werden die Faktoren w^0 , w^1 und w^2 auch als Twiddle-Faktoren bezeichnet.

30

Fig. 3 zeigt einen FFT-Algorithmus mit einer veränderten Abfolge der Rechenoperationen, den sog. Singleton-Algorithmus, der in der oben genannten Literaturstelle Oppenheimer-Schafer auf S. 301 beschrieben ist, und der in dem erfindungsgemäßen

Eine, in der Fig. 5 dargestellt zweite Ausführungsform zur schnellen Berechnung einer DIT-Transformation wie die der Fig. 3 ist die Verwendung eines kleinen schnellen Speichers 5, der nach der Butterfly 2 angeordnet ist. Dort werden einige Zwischenergebnisse der Berechnung zwischengespeichert, um sie dann ohne ständigen Seitenwechsel in den seitenorientierten Speicher 1 zu schreiben. Die Berechnung verläuft nun, indem aus dem Speicher 1 zwei Daten gelesen, der Butterfly zugeführt und die Zwischenergebnisse im schnellen SRAM-Speicher 5 (SRAM = Statisches RAM) gespeichert werden. Die Zwischenergebnisse werden dann seitenweise in den seitenorientierten Speicher 1 geschrieben.

Bei einer Berechnung nach dem DIF-Verfahren sitzt der statische schnelle Speicher 5 am Eingang der Butterfly 2 (nicht dargestellt). Ansonsten ist die Funktionsweise analog zu denjenigen der erläuterten DIT-Variante.

Die folgende Tabelle 1 soll das Adressierungsschema der Berechnung mittels Zwischenspeicherung verdeutlichen:

Tabelle 1

DRAM (RD)	Butterfly	SRAM (WR)	SRAM (RD)	DRAM (WR)
0	0	0		
1	16	8		
2	1	1		
3	17	9		
4	2	2	0	0
5	18	10	1	1
6	3	3	2	2
7	19	11	3	3
8	4	4	8	16
9	20	12	9	17
10	5	5	10	18
11	21	13	11	19
12	6	6	4	4

Daten seitenweise linear ausgelesen, also in der Tabelle 1 unter SRAM (RD) die Adressen 0-3, anschließend 8-11, usw. Der Inhalt dieser SRAM-Adressen wird in die entsprechenden, durch die Transformation bedingten Adressen des langsamsten seitenorientierten Speichers (Spalte DRAM (WR)) seitenweise linear geschrieben, also im Beispiel nach 0-3, 16-19, 4-7 usw. Durch die Verwendung des kleinen schnellen Speichers wird daher der Zugriff auf den seitenorientierten Speicher DRAM derart optimiert, daß möglichst wenige langsame Seitenwechsel notwendig sind.

Die obengenannten und erläuterten Transformationen sowie Speichergrößen dienen nur zur Erläuterung. In der Praxis ist der langsame Speicher wesentlich größer als der schnelle Speicher. Beispielsweise ist üblicherweise der langsame Speicher für $N = 8192$ ausgelegt, wobei der langsame Speicher eine Seitengröße von $P = 16$ aufweist. Daher hat der kleine Speicher eine Größe von 64 Adressen. In einer integrierten Realisierung fällt der schnelle kleine Speicher daher flächenmäßig kaum ins Gewicht, allerdings wird die Berechnung der FFT oder IFFT oder ähnlicher diskreter orthogonaler Transformationen durch die Minimierung der Seitenwechsel des langsamsten Speichers erheblich beschleunigt.

Bezugszeichenliste

1	seitenorientierter Speicher
2	Butterfly
3	seitenorientierter Speicher
30 4	seitenorientierter Speicher
5	schneller Speicher

die resultierenden Daten wieder seitenweise in die beiden seitenweise organisierten Speicher (3, 4) geschrieben werden.

4. Verfahren nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, daß die diskrete orthogonale Transformation durch eine FFT, IFFT, DCT oder IDCT gebildet wird.

5. Verfahren nach Anspruch 4, dadurch gekennzeichnet, daß die Transformation eine identische Geometrie für 10 jede Stufe aufweist.

6. Verfahren nach Anspruch 5, dadurch gekennzeichnet, daß eine FFT oder IFFT nach Singleton verwendet wird.

15 7. Vorrichtung zur Durchführung des Verfahrens nach einem der Ansprüche 1, 4-6,

dadurch gekennzeichnet, daß die Vorrichtung einen seitenweise organisierten Speicher (1), eine Recheneinheit (2) und einen nach der Recheneinheit angeordneten direkt organisierten Speicher (5) aufweist.

8. Vorrichtung zur Durchführung des Verfahrens nach einem der Ansprüche 2, 4-6,

dadurch gekennzeichnet, daß die Vorrichtung einen seitenweise organisierten Speicher (1), eine Recheneinheit (2) und einen vor der Recheneinheit angeordneten direkt organisierten Zwischenspeicher (5) aufweist.

30 9. Vorrichtung nach einem der Ansprüche 7 oder 8, dadurch gekennzeichnet, daß der seitenorientierte Speicher (1) im Verhältnis zu dem direkt organisierten Zwischenspeicher (5) ein großer Speicher ist.

35 10. Vorrichtung nach Anspruch 9, dadurch gekennzeichnet, daß für den Zwischenspeicher (5) ein schneller Speicher verwendet wird.

Zusammenfassung

Verfahren und Vorrichtung zum Berechnen einer diskreten orthogonalen Transformation wie FFT oder IFFT

5

Ein Verfahren zum Berechnen einer orthogonalen diskreten Transformation nach dem DIT-Verfahren in vorgegebenen Zwischenschritten weist die folgenden Schritte auf:

- a) Lesen der Daten aus einem Seitenweise organisierten Speicher,
- b) Durchführen der durch die Transformation vorgegebenen Zwischenschritt,
- c) Speichern der resultierenden Daten in einen Zwischen-Speicher, und
- d) Seitenweises Schreiben der resultierenden Daten aus dem Zwischenspeicher in den Seitenweise organisierten Speicher.

Als diskrete orthogonale Transformationen kommen FFT, IFFT, DCT, IDCT und strukturell ähnliche Transformationen in Frage.

20

Fig. 4

1 / 2

Fig. 1

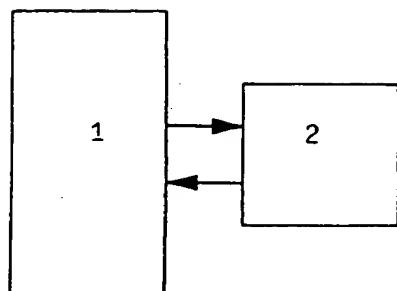
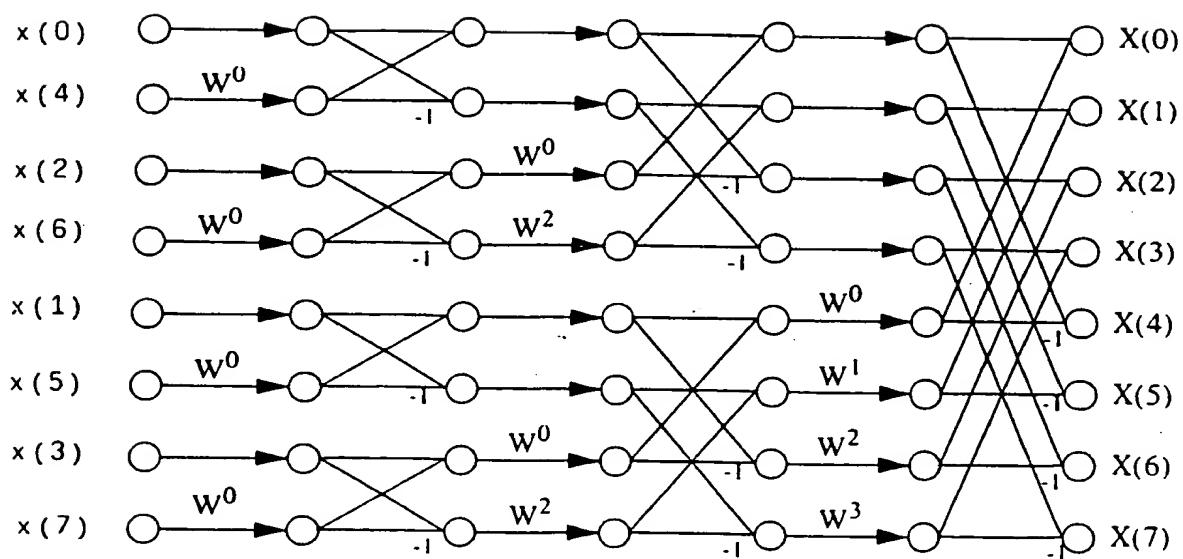


Fig. 2



2 / 2

Fig. 3

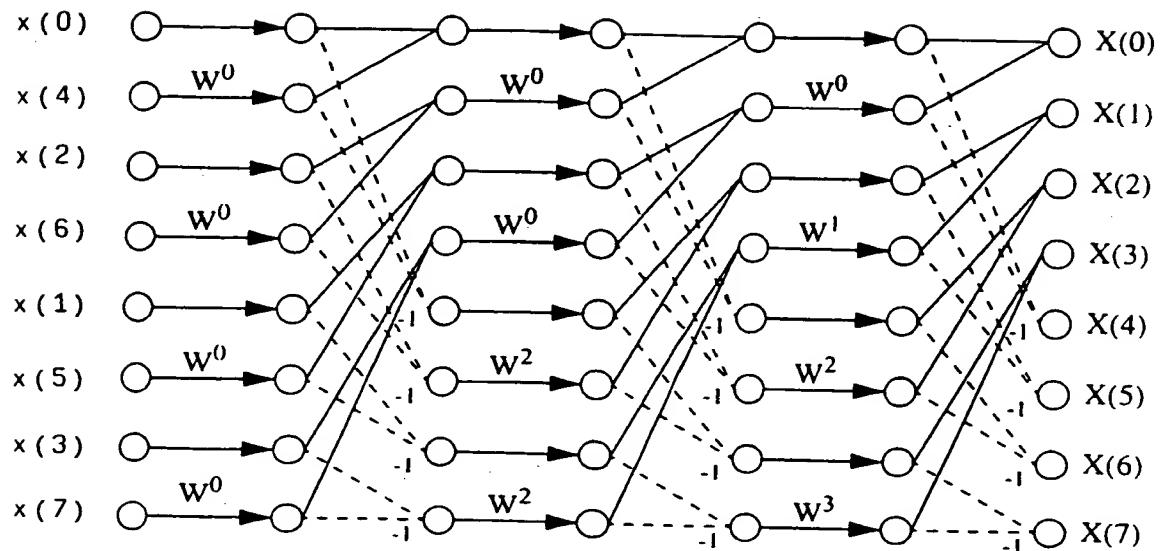


Fig. 4

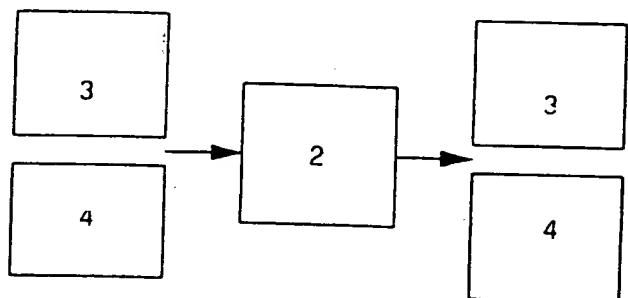


Fig. 5

